EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

As demonstrated in the chart below, Hisense directly and indirectly infringes at least claim 9 of US 10,536,714 (the "'714 Patent"). Hisense directly infringes, contributes to the infringement of, and/or induces infringement of the '714 Patent by making, using, selling, offering for sale, and/or importing into the United States the Accused Products that are covered by one or more claims of the '714 Patent. The Accused Products are devices that decode H.265-compliant video. For example, the Hisense 43A7N is a representative product for other Hisense devices that decode H.265-compliant video.

The Hisense 43A7N contains at least one video decoder that helps decode H.265-compliant video.[1] While evidence from the Hisense 43A7N is specifically charted herein, the evidence and contentions charted herein apply equally to the other Hisense Accused Products that decode H.265-compliant video.

No part of this exemplary chart construes, or is intended to construe, the specification, file history, or claims of the '714 Patent. Moreover, this exemplary chart does not limit, and is not intended to limit, Nokia's infringement positions or contentions.

The following infringement chart includes exemplary citations to ITU-T Rec. H.265 (12/2016) High efficiency video coding (available at https://www.itu.int/rec/T-REC-H.265-201612-S/en) (the "H.265 Standard"). The cited functionality has been included in editions of the H.265 Standard since April 2013 and remains in current editions of the H.265 Standard. Any Hisense device that includes a decoder that practices the functionality in any of these editions of the H.265 Standard ("H.265 Decoder") practices claims of the '714 Patent. Thus, the Accused Products each practice the H.265 Standard and are covered by claims of the '714 Patent.

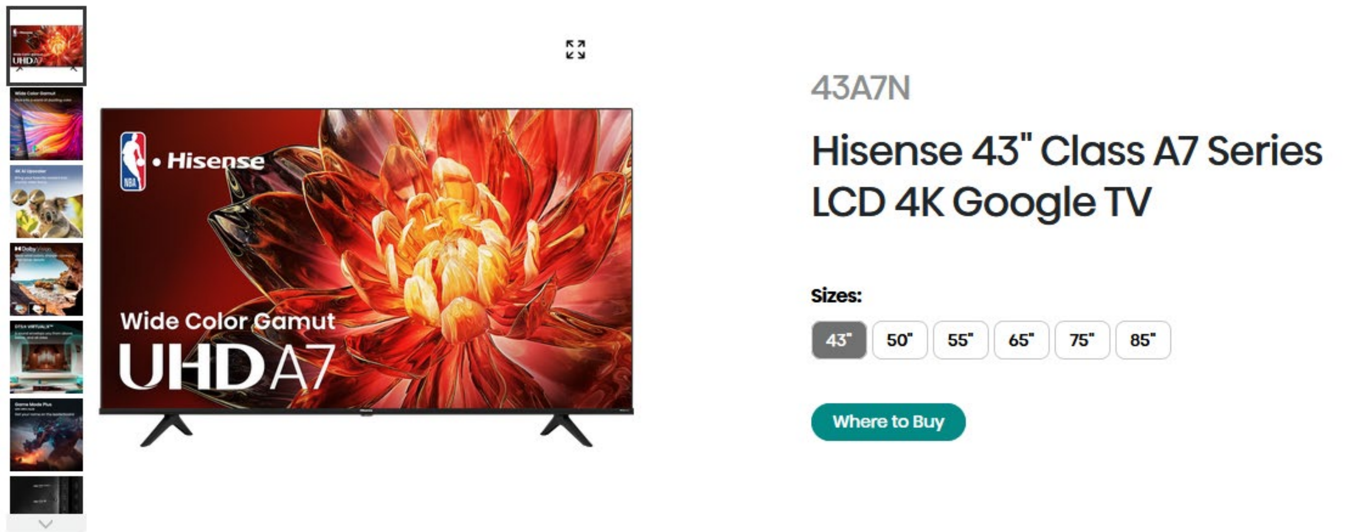
Nokia contends each of the following limitations is met literally, and, to the extent a limitation is not met literally, it is met under the doctrine of equivalents.[2]

[1] See, e.g., Hisense 43A7N User Manual available at https://www.hisense-usa.com/televisions/hisense-43-class-a7-series-4k-wide-color-gamut-google-tv-43a7n.

[2] This claim chart is based on the information currently available to Nokia and is intended to be exemplary in nature. Nokia reserves all rights to update and elaborate its infringement positions, including as Nokia obtains additional information during discovery.

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

<table>
<tr><th>U.S. PATENT NO. 10,536,714</th><th>HISENSE ACCUSED PRODUCTS</th></tr>
<tr><td>9. [A] A method comprising:</td><td>
Each of the Accused Products, such as the Hisense 43A7N, performs a method comprising the limitations below.
For example, and without limitation, the Hisense 43A7N uses hardware-accelerated video decoding and includes Mediatek MT9602 Processor.



https://www.hisense-usa.com/televisions/hisense-43-class-a7-series-4k-wide-color-gamut-google-tv-43a7n (last accessed March 29, 2025).
</td></tr>
</table>

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	 **Display**: 42.5 in, VA, Direct LED, 3840 x 2160 pixels **Viewing angles (H/V)**: 178 ° / 178 ° **Brightness**: 275 cd/m² **Static contrast**: 4000 : 1 **Refresh rate**: 50 Hz / 60 Hz **Frame interpolation**: 120 MR (Motion Rate) **TV tuner**: Analog (NTSC/PAL/SECAM), ATSC, Clear QAM **SoC**: MediaTek MT9602 **CPU**: ARM Cortex-A53, 1500 MHz, **Cores**: 4 **Dimensions**: 963 x 560 x 74 mm **Weight**: 6.8 kg Add to compare Suggest an edit https://www.displayspecifications.com/en/model/234a3f3b

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	Video Format

Container	Video Codec	File Extension Name	Resolution and Frame Rate
MPEG program stream	MPEG1/2	.DAT, .VOB, .MPG, .MPEG	1920 x 1080 @ 60fps
	MPEG4		
	H.264		3840 x 2160 @ 60fps
MPEG transport stream	HEVC/H.265	.ts, .trp, .tp	3840 x 2160 @ 60fps
	MPEG4		1920 x 1080 @ 60fps
	H.264		3840 x 2160 @ 60fps
	VC1		1920 x 1080 @ 60fps

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	(see table below)

Container	Video Codec	File Extension Name	Resolution and Frame Rate
	MPEG1/2		1920 x 1080 @ 60fps
	AVS		
	AVS+		
	AVS2		3840 x 2160 @ 60fps
MP4	VP8	.mp4, .mov	1920 x 1080 @ 60fps
	AV1		3840 x 2160 @ 60fps
	HEVC/H.265		
	MPEG1/2		1920 x 1080 @ 60fps
	MPEG4		
	H.263		
	H.264		3840 x 2160 @ 60fps
	WMV3		1920 x 1080 @ 60fps
	VC1		
	Motion JPEG		1920 x 1080 @ 30fps

Source: Hisense 43A7N User Manual, at 45-46. Downloaded from https://www.hisense-usa.com/televisions/hisense-43-class-a7-series-4k-wide-color-gamut-google-tv-43a7n (last accessed March 29, 2025).

For example, an Hisense 43A7N was used to playback an H.265-compliant video.

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

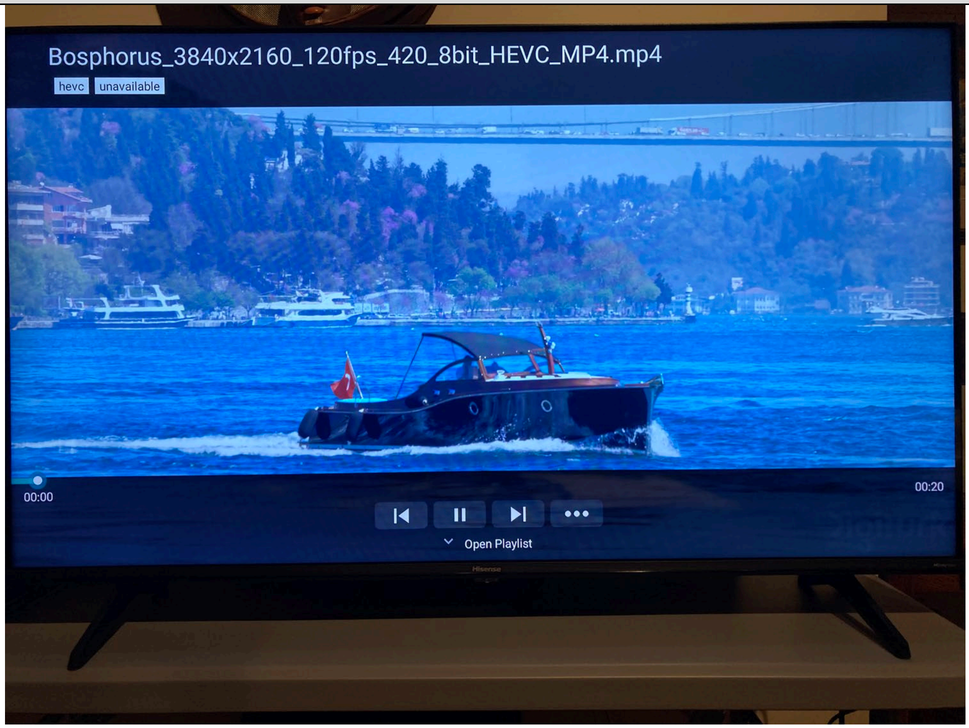
U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	 Source: Picture of H.265-complaint video playback on Hisense 43A7N For example, and without limitation, the H.265 Standard specifies the following regarding the decoding process. Each of the Hisense Accused Products performs a method comprising the limitations below. **3 Definitions** For the purposes of this Recommendation | International Standard, the following definitions apply. . . .

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	3.12 bitstream: A sequence of bits, **. . .** , that forms the representation of *coded pictures* and associated data forming one or more coded video sequences *(CVSs)*. . . . **3.25 coded picture**: A *coded representation* of a *picture* **. . .** . . . **3.44 decoding process**: The process specified in this Specification that reads a *bitstream* and derives *decoded pictures* from it. ITU-T Rec. H.265 (12/2016) High efficiency video coding at pp. 4 – 7.
[B] selecting a first spatial motion vector prediction candidate from a set of spatial motion vector prediction candidates for an encoded block of pixels as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list for a prediction unit of the encoded block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates;	Each of the Accused Products, such as the Hisense 43A7N, performs a method comprising selecting a first spatial motion vector prediction candidate from a set of spatial motion vector prediction candidates for an encoded block of pixels as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list for a prediction unit of the encoded block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates. For example, and without limitation, the H.265 Standard specifies the following regarding the decoding process. The Accused Products perform a method comprising selecting a first spatial motion vector prediction candidate from a set of spatial motion vector prediction candidates for an encoded block of pixels as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list for a prediction unit of the encoded block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates, corresponding to the decoding process specified by the H.265 Standard. As specified in Subclause 8.5.3.2.3 of the H.265 Standard, the spatial motion vector prediction candidates A0, A1, B0, B1, and B2 (see Figure 8-3) are processed in the order of A1, B1, B0, A0, and B2. For each location A1, B1, B0, A0, and B2, the Accused Products check the availability of the block as specified in Subclause 6.4.2. If a block is coded in intra-coding or not available (e.g., the block is outside of the current

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	slice or tile), its availability flag is set to FALSE, and it is not considered as a candidate for motion vector prediction and it is not added to the candidate list. The following specifications provide further evidence of how each of the Accused Products operates: **8.5.3.2.3 Derivation process for spatial merging candidates** . . . For the derivation of availableFlagA_1, refIdxLXA_1, predFlagLXA_1 and mvLXA_1 the following applies: – The luma location (xNbA_1, yNbA_1) inside the neighbouring luma coding block is set equal to (xPb − 1, yPb + nPbH). – The availability derivation process for a prediction block as specified in clause 6.4.2 is invoked with the luma location (xCb, yCb), the current luma coding block size nCbS, the luma prediction block location (xPb, yPb), the luma prediction block width nPbW, the luma prediction block height nPbH, the luma location (xNbA_1, yNbA_1) and the partition index partIdx as inputs, and the output is assigned to the prediction block availability flag availableA_1. . . . For the derivation of availableFlagB_1, refIdxLXB_1, predFlagLXB_1 and mvLXB_1 the following applies: – The luma location (xNbB_1, yNbB_1) inside the neighbouring luma coding block is set equal to (xPb + nPbW − 1, yPb − 1). – The availability derivation process for a prediction block as specified in clause 6.4.2 is invoked with the luma location (xCb, yCb), the current luma coding block size nCbS, the luma prediction block location (xPb, yPb), the luma prediction block width nPbW, the luma prediction block height nPbH, the luma location (xNbB_1, yNbB_1) and the partition index partIdx as inputs, and the output is assigned to the prediction block availability flag availableB_1.

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	. . . For the derivation of availableFlagB_0, refIdxLXB_0, predFlagLXB_0 and mvLXB_0 the following applies: – The luma location (xNbB_0, yNbB_0) inside the neighbouring luma coding block is set equal to (xPb + nPbW, yPb − 1). – The availability derivation process for a prediction block as specified in clause 6.4.2 is invoked with the luma location (xCb, yCb), the current luma coding block size nCbS, the luma prediction block location (xPb, yPb), the luma prediction block width nPbW, the luma prediction block height nPbH, the luma location (xNbB_0, yNbB_0) and the partition index partIdx as inputs, and the output is assigned to the prediction block availability flag availableB_0. . . . For the derivation of availableFlagA0, refIdxLXA0, predFlagLXA0 and mvLXA0 the following applies: – The luma location (xNbA0, yNbA0) inside the neighbouring luma coding block is set equal to (xPb − 1, yPb + nPbH). – The availability derivation process for a prediction block as specified in clause 6.4.2 is invoked with the luma location (xCb, yCb), the current luma coding block size nCbS, the luma prediction block location (xPb, yPb), the luma prediction block width nPbW, the luma prediction block height nPbH, the luma location (xNbA0, yNbA0) and the partition index partIdx as inputs, and the output is assigned to the prediction block availability flag availableA0. . . . For the derivation of availableFlagB2, refIdxLXB2, predFlagLXB2 and mvLXB2 the following applies:

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

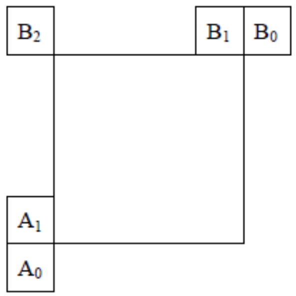
U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	– The luma location (xNbB2, yNbB2) inside the neighbouring luma coding block is set equal to (xPb − 1, yPb − 1). – The availability derivation process for a prediction block as specified in clause 6.4.2 is invoked with the luma location (xCb, yCb), the current luma coding block size nCbS, the luma prediction block location (xPb, yPb), the luma prediction block width nPbW, the luma prediction block height nPbH, the luma location (xNbB2, yNbB2) and the partition index partIdx as inputs, and the output is assigned to the prediction block availability flag availableB2. ITU-T Rec. H.265 (12/2016) High efficiency video coding at pp. 150-152. **8.5.3.2.7 Derivation process for motion vector predictor candidates** . . .  **Figure 8-3 – Spatial motion vector neighbours (informative)** ITU-T Rec. H.265 (12/2016) High efficiency video coding at pp. 156.
[C] determining a subset of spatial motion vector prediction candidates based on the location of the block associated with the first	Each of the Accused Products, such as the Hisense 43A7N, performs a method comprising determining a subset of spatial motion vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate.

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
spatial motion vector prediction candidate;	For example, the spatial motion vector prediction candidates A1, B1, B0, A0, and B2 are processed. For example, when the Accused Products select spatial motion vector prediction candidate at position B2 as a potential candidate, they determine a subset of candidates as (B1, A1). *See* Steps 8 and 9 in Subclause 8.5.3.2.3. In another example, when B0 is selected, the Accused Product determine the subset as (B1). *See* Step 4 in Subclause 8.5.3.2.3. In yet another example, when A0 is selected, the Accused Products determine the subset as (A1). *See* Step 6 in Subclause 8.5.3.2.3; *see* Subclause 8.5.3.2.3. **8.5.3.2.3 Derivation process for spatial merging candidates** . . . For the derivation of availableFlagB_1, refIdxLXB_1, predFlagLXB_1 and mvLXB_1 the following applies: – The luma location (xNbB_1, yNbB_1) inside the neighbouring luma coding block is set equal to (xPb + nPbW − 1, yPb − 1). . . . – The variables availableFlagB_1, refIdxLXB_1, predFlagLXB_1 and mvLXB_1 are derived as follows: – If one or more of the following conditions are true, availableFlagB_1 is set equal to 0, both components of mvLXB_1 are set equal to 0, refIdxLXB_1 is set equal to −1 and predFlagLXB_1 is set equal to 0, with X being 0 or 1: . . . 2. availableA_1 is equal to TRUE and the prediction units covering the luma locations (xNbA_1, yNbA_1) and (xNbB_1, yNbB_1) have the same motion vectors and the same reference indices. . . . For the derivation of availableFlagB_0, refIdxLXB_0, predFlagLXB_0 and mvLXB_0 the following applies: – The luma location (xNbB_0, yNbB_0) inside the neighbouring luma coding block is set equal to (xPb + nPbW, yPb − 1).

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	. . . – The variables availableFlagB_0, refIdxLXB_0, predFlagLXB_0 and mvLXB_0 are derived as follows: – If one or more of the following conditions are true, availableFlagB_0 is set equal to 0, both components of mvLXB_0 are set equal to 0, refIdxLXB_0 is set equal to −1 and predFlagLXB_0 is set equal to 0, with X being 0 or 1: . . . 4. availableB_1 is equal to TRUE and the prediction units covering the luma locations (xNbB_1, yNbB_1) and (xNbB_0, yNbB_0) have the same motion vectors and the same reference indices. . . . For the derivation of availableFlagA_0, refIdxLXA_0, predFlagLXA_0 and mvLXA_0 the following applies: – The luma location (xNbA_0, yNbA_0) inside the neighbouring luma coding block is set equal to (xPb − 1, yPb + nPbH). . . . – The variables availableFlagA_0, refIdxLXA_0, predFlagLXA_0 and mvLXA_0 are derived as follows: – If one or more of the following conditions are true, availableFlagA_0 is set equal to 0, both components of mvLXA_0 are set equal to 0, refIdxLXA_0 is set equal to −1 and predFlagLXA_0 is set equal to 0, with X being 0 or 1: . . . 6. availableA_1 is equal to TRUE and the prediction units covering the luma locations (xNbA_1, yNbA_1) and (xNbA_0, yNbA_0) have the same motion vectors and the same reference indices. . . . For the derivation of availableFlagB_2, refIdxLXB_2, predFlagLXB_2 and mvLXB_2 the following applies:

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	– The luma location ($xNbB_2$, $yNbB_2$) inside the neighbouring luma coding block is set equal to ($xPb - 1$, $yPb - 1$). . . . – The variables $availableFlagB_2$, $refIdxLXB_2$, $predFlagLXB_2$ and $mvLXB_2$ are derived as follows: – If one or more of the following conditions are true, $availableFlagB_2$ is set equal to 0, both components of $mvLXB_2$ are set equal to 0, $refIdxLXB_2$ is set equal to −1 and $predFlagLXB_2$ is set equal to 0, with X being 0 or 1: . . . 8. $availableA_1$ is equal to TRUE and prediction units covering the luma locations ($xNbA_1$, $yNbA_1$) and ($xNbB_2$, $yNbB_2$) have the same motion vectors and the same reference indices. 9. $availableB_1$ is equal to TRUE and the prediction units covering the luma locations ($xNbB_1$, $yNbB_1$) and ($xNbB_2$, $yNbB_2$) have the same motion vectors and the same reference indices. . . . . . . ITU-T Rec. H.265 (12/2016) High efficiency video coding at pp. 150-152. **8.5.3.2.7 Derivation process for motion vector predictor candidates** . . .

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

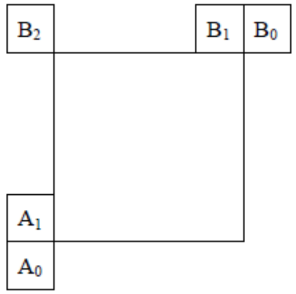
U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	 **Figure 8-3 – Spatial motion vector neighbours (informative)** ITU-T Rec. H.265 (12/2016) High efficiency video coding at p. 156
[D] comparing motion information of the first spatial motion vector prediction candidate with motion information of another spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates;	Each of the Accused Products, such as the Hisense 43A7N, performs a method comprising comparing motion information of the first spatial motion vector prediction candidate with motion information of another spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates. For example, when considering the spatial motion vector prediction candidate at position B2, the Accused Products compare motion information for spatial motion vector prediction candidate at position B2 with motion information of spatial motion vector prediction candidates in the subset (B1, A1) of spatial motion vector prediction candidates. *See* Steps 8 and 9 in Subclause 8.5.3.2.3. Motion information of spatial motion vector prediction candidate at position B2 is not compared with motion information of A0, and B0. In this example, the Accused Products checks whether motion information at position B2 is equal to motion information at position B1 and motion information at position A1. *See* Steps 8 and 9 in Subclause 8.5.3.2.3. If motion information at position B2 is equal to motion information at either position B1 or A1, then B2 will not be included in the list. As another example, the Accused Products compare motion information for spatial motion vector prediction candidate at position A0 with motion information of spatial motion vector prediction candidates in the subset

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	(A1) of spatial motion vector prediction candidates. *See* Step 6 in Subclause 8.5.3.2.3. Motion information of spatial motion vector prediction candidate at position A0 is not compared with motion information of B1, B0, and B2. In this example, the Accused Products check whether motion information at position A0 is equal to motion information at position A1. *See* Step 6 in Subclause 8.5.3.2.3. If motion information at position A0 is equal to motion information at position A1, then A0 will not be included in the list. **8.5.3.2.3 Derivation process for spatial merging candidates** . . . For the derivation of availableFlagB_1, refIdxLXB_1, predFlagLXB_1 and mvLXB_1 the following applies: – The luma location (xNbB_1, yNbB_1) inside the neighbouring luma coding block is set equal to (xPb + nPbW − 1, yPb − 1). . . . – The variables availableFlagB_1, refIdxLXB_1, predFlagLXB_1 and mvLXB_1 are derived as follows: – If one or more of the following conditions are true, availableFlagB_1 is set equal to 0, both components of mvLXB_1 are set equal to 0, refIdxLXB_1 is set equal to −1 and predFlagLXB_1 is set equal to 0, with X being 0 or 1: . . . 1. availableB_1 is equal to FALSE. 2. availableA_1 is equal to TRUE and the prediction units covering the luma locations (xNbA_1, yNbA_1) and (xNbB_1, yNbB_1) have the same motion vectors and the same reference indices. – Otherwise, availableFlagB_1 is set equal to 1 and the following assignments are made: mvLXB1 = MvLX[xNbB1][yNbB1] (8-131) refIdxLXB1 = RefIdxLX[xNbB1][yNbB1] (8-132)

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	predFlagLXB1 = PredFlagLX[xNbB1][yNbB1] (8-133) For the derivation of availableFlagB_0, refIdxLXB_0, predFlagLXB_0 and mvLXB_0 the following applies: – The luma location (xNbB_0, yNbB_0) inside the neighbouring luma coding block is set equal to (xPb + nPbW, yPb − 1). . . . – The variables availableFlagB_0, refIdxLXB_0, predFlagLXB_0 and mvLXB_0 are derived as follows: – If one or more of the following conditions are true, availableFlagB_0 is set equal to 0, both components of mvLXB_0 are set equal to 0, refIdxLXB_0 is set equal to −1 and predFlagLXB_0 is set equal to 0, with X being 0 or 1: . . . 3. availableB_0 is equal to FALSE. 4. availableB_1 is equal to TRUE and the prediction units covering the luma locations (xNbB_1, yNbB_1) and (xNbB_0, yNbB_0) have the same motion vectors and the same reference indices. – Otherwise, availableFlagB_0 is set equal to 1 and the following assignments are made: mvLXB0 = MvLX[xNbB0][yNbB0] (8-134) refIdxLXB0 = RefIdxLX[xNbB0][yNbB0] (8-135) predFlagLXB0 = PredFlagLX[xNbB0][yNbB0] (8-136) For the derivation of availableFlagA0, refIdxLXA0, predFlagLXA0 and mvLXA0 the following applies: – The luma location (xNbA0, yNbA0) inside the neighbouring luma coding block is set equal to (xPb − 1, yPb + nPbH).

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	. . . – The variables availableFlagA0, refIdxLXA0, predFlagLXA0 and mvLXA0 are derived as follows: – If one or more of the following conditions are true, availableFlagA0 is set equal to 0, both components of mvLXA0 are set equal to 0, refIdxLXA0 is set equal to −1 and predFlagLXA0 is set equal to 0, with X being 0 or 1: 5. availableA0 is equal to FALSE. 6. availableA1 is equal to TRUE and the prediction units covering the luma locations (xNbA1, yNbA1) and (xNbA0, yNbA0) have the same motion vectors and the same reference indices. – Otherwise, availableFlagA0 is set equal to 1 and the following assignments are made: mvLXA0 = MvLX[xNbA0][yNbA0] (8-137) refIdxLXA0 = RefIdxLX[xNbA0][yNbA0] (8-138) predFlagLXA0 = PredFlagLX[xNbA0][yNbA0] (8-139) For the derivation of availableFlagB2, refIdxLXB2, predFlagLXB2 and mvLXB2 the following applies: – The luma location (xNbB2, yNbB2) inside the neighbouring luma coding block is set equal to (xPb − 1, yPb − 1). . . . – The variables availableFlagB2, refIdxLXB2, predFlagLXB2 and mvLXB2 are derived as follows:

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	– If one or more of the following conditions are true, availableFlagB2 is set equal to 0, both components of mvLXB2 are set equal to 0, refIdxLXB2 is set equal to −1 and predFlagLXB2 is set equal to 0, with X being 0 or 1: 7. availableB2 is equal to FALSE. 8. availableA1 is equal to TRUE and prediction units covering the luma locations (xNbA1, yNbA1) and (xNbB2, yNbB2) have the same motion vectors and the same reference indices. 9. availableB1 is equal to TRUE and the prediction units covering the luma locations (xNbB1, yNbB1) and (xNbB2, yNbB2) have the same motion vectors and the same reference indices. 10. availableFlagA0 + availableFlagA1 + availableFlagB0 + availableFlagB1 is equal to 4. – Otherwise, availableFlagB2 is set equal to 1 and the following assignments are made: mvLXB2 = MvLX[xNbB2][yNbB2] (8-140) refIdxLXB2 = RefIdxLX[xNbB2][yNbB2] (8-141) predFlagLXB2 = PredFlagLX[xNbB2][yNbB2] (8-142) ITU-T Rec. H.265 (12/2016) High efficiency video coding at pp. 150-152. **8.5.3.2.7 Derivation process for motion vector predictor candidates** . . .

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

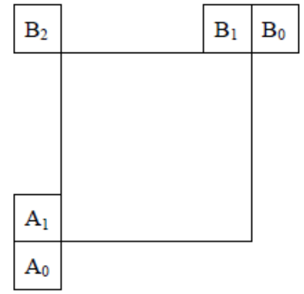
U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	 **Figure 8-3 – Spatial motion vector neighbours (informative)** ITU-T Rec. H.265 (12/2016) High efficiency video coding at p. 156.
[E] determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing; and	Each of the Accused Products, such as the Hisense 43A7N, performs a method comprising determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing. For example, and as explained above, when considering the spatial motion vector prediction candidate at position B2, the Accused Products compare motion information for spatial motion vector prediction candidate at position B2 with motion information of spatial motion vector prediction candidates in the subset (B1, A1) of spatial motion vector prediction candidates. *See* Steps 8 and 9 in Subclause 8.5.3.2.3. Motion information of spatial motion vector prediction candidate at position B2 is not compared with motion information of A0, and B0. In this example, the Accused Products checks whether motion information at position B2 is equal to motion information at position B1 and motion information at position A1. *See* Steps 8 and 9 in Subclause 8.5.3.2.3. If motion information at position B2 is equal to motion information at either position B1 or A1, then B2 will not be included in the list. As another example, and as also explained above, the Accused Products compare motion information for spatial motion vector prediction candidate at position A0 with motion information of spatial motion vector prediction candidates in the subset (A1) of spatial motion vector prediction candidates. *See* Step 6 in Subclause 8.5.3.2.3. Motion information of spatial motion vector prediction candidate at position A0 is not

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	compared with motion information of B1, B2, and B0. In this example, the Accused Products check whether motion information at position A0 is equal to motion information at position A1. *See* Step 6 in Subclause 8.5.3.2.3. If motion information at position A0 is equal to motion information at position A1, then A0 will not be included in the list. **8.5.3.2.3 Derivation process for spatial merging candidates** . . . For the derivation of availableFlagB_1, refIdxLXB_1, predFlagLXB_1 and mvLXB_1 the following applies: – The luma location (xNbB_1, yNbB_1) inside the neighbouring luma coding block is set equal to (xPb + nPbW − 1, yPb − 1). . . . – The variables availableFlagB_1, refIdxLXB_1, predFlagLXB_1 and mvLXB_1 are derived as follows: – If one or more of the following conditions are true, availableFlagB_1 is set equal to 0, both components of mvLXB_1 are set equal to 0, refIdxLXB_1 is set equal to −1 and predFlagLXB_1 is set equal to 0, with X being 0 or 1: . . . 5. availableB_1 is equal to FALSE. 6. availableA_1 is equal to TRUE and the prediction units covering the luma locations (xNbA_1, yNbA_1) and (xNbB_1, yNbB_1) have the same motion vectors and the same reference indices. – Otherwise, availableFlagB_1 is set equal to 1 and the following assignments are made: mvLXB1 = MvLX[xNbB1][yNbB1] (8-131) refIdxLXB1 = RefIdxLX[xNbB1][yNbB1] (8-132) predFlagLXB1 = PredFlagLX[xNbB1][yNbB1] (8-133)

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	For the derivation of availableFlagB_0, refIdxLXB_0, predFlagLXB_0 and mvLXB_0 the following applies: – The luma location (xNbB_0, yNbB_0) inside the neighbouring luma coding block is set equal to (xPb + nPbW, yPb − 1). . . . – The variables availableFlagB_0, refIdxLXB_0, predFlagLXB_0 and mvLXB_0 are derived as follows: – If one or more of the following conditions are true, availableFlagB_0 is set equal to 0, both components of mvLXB_0 are set equal to 0, refIdxLXB_0 is set equal to −1 and predFlagLXB_0 is set equal to 0, with X being 0 or 1: . . . 7. availableB_0 is equal to FALSE. 8. availableB_1 is equal to TRUE and the prediction units covering the luma locations (xNbB_1, yNbB_1) and (xNbB_0, yNbB_0) have the same motion vectors and the same reference indices. – Otherwise, availableFlagB_0 is set equal to 1 and the following assignments are made: mvLXB0 = MvLX[xNbB0][yNbB0] (8-134) refIdxLXB0 = RefIdxLX[xNbB0][yNbB0] (8-135) predFlagLXB0 = PredFlagLX[xNbB0][yNbB0] (8-136) For the derivation of availableFlagA0, refIdxLXA0, predFlagLXA0 and mvLXA0 the following applies: – The luma location (xNbA0, yNbA0) inside the neighbouring luma coding block is set equal to (xPb − 1, yPb + nPbH). . . .

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	– The variables availableFlagA0, refIdxLXA0, predFlagLXA0 and mvLXA0 are derived as follows: – If one or more of the following conditions are true, availableFlagA0 is set equal to 0, both components of mvLXA0 are set equal to 0, refIdxLXA0 is set equal to −1 and predFlagLXA0 is set equal to 0, with X being 0 or 1: 5. availableA0 is equal to FALSE. 6. availableA1 is equal to TRUE and the prediction units covering the luma locations (xNbA1, yNbA1) and (xNbA0, yNbA0) have the same motion vectors and the same reference indices. – Otherwise, availableFlagA0 is set equal to 1 and the following assignments are made: mvLXA0 = MvLX[xNbA0][yNbA0] (8-137) refIdxLXA0 = RefIdxLX[xNbA0][yNbA0] (8-138) predFlagLXA0 = PredFlagLX[xNbA0][yNbA0] (8-139) For the derivation of availableFlagB2, refIdxLXB2, predFlagLXB2 and mvLXB2 the following applies: – The luma location (xNbB2, yNbB2) inside the neighbouring luma coding block is set equal to (xPb − 1, yPb − 1). . . . – The variables availableFlagB2, refIdxLXB2, predFlagLXB2 and mvLXB2 are derived as follows: – If one or more of the following conditions are true, availableFlagB2 is set equal to 0, both components of mvLXB2 are set equal to 0, refIdxLXB2 is set equal to −1 and predFlagLXB2 is set equal to 0, with X being 0 or 1:

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

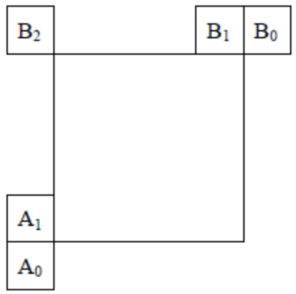
U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	7. availableB2 is equal to FALSE. 8. availableA1 is equal to TRUE and prediction units covering the luma locations (xNbA1, yNbA1) and (xNbB2, yNbB2) have the same motion vectors and the same reference indices. 9. availableB1 is equal to TRUE and the prediction units covering the luma locations (xNbB1, yNbB1) and (xNbB2, yNbB2) have the same motion vectors and the same reference indices. 10. availableFlagA0 + availableFlagA1 + availableFlagB0 + availableFlagB1 is equal to 4. – Otherwise, availableFlagB2 is set equal to 1 and the following assignments are made: mvLXB2 = MvLX[xNbB2][yNbB2] (8-140) refIdxLXB2 = RefIdxLX[xNbB2][yNbB2] (8-141) predFlagLXB2 = PredFlagLX[xNbB2][yNbB2] (8-142) ITU-T Rec. H.265 (12/2016) High efficiency video coding at pp. 150-152. **8.5.3.2.7 Derivation process for motion vector predictor candidates** . . .  **Figure 8-3 – Spatial motion vector neighbours (informative)** ITU-T Rec. H.265 (12/2016) High efficiency video coding at p. 156.

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
[F] selecting a spatial motion vector prediction candidate from the motion vector prediction list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the motion vector prediction list using information that was received identifying a respective spatial motion vector prediction candidate from the motion vector prediction list constructed by an encoder.	Each of the Accused Products, such as the Hisense 43A7N, performs a method comprising selecting a spatial motion vector prediction candidate from the motion vector prediction list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the motion vector prediction list using information that was received identifying a respective spatial motion vector prediction candidate from the motion vector prediction list constructed by an encoder. For example, the Accused Products receive syntax element merge_idx from the bitstream. *See* Subclause 7.3.8.6. The Accused Products select the motion vector prediction candidate at position merge_idx in the merging candidate list mergeCandList. *See* Step 9 in Subclause 8.5.3.2.2. The Accused Products then use that motion vector in decoding the current block. **3 Definitions** For the purposes of this Recommendation | International Standard, the following definitions apply. . . . **3.153 syntax element**: An element of data represented in the *bitstream*. **3.154 syntax structure**: Zero or more *syntax elements* present together in the *bitstream* in a specified order. ITU-T Rec. H.265 (12/2016) High efficiency video coding at pp. 5, 7-12. **5.10 Variables, syntax elements and tables** Syntax elements in the bitstream are represented in **bold** type. Each syntax element is described by its name (all lower case letters with underscore characters), and one descriptor for its method of coded representation. The decoding process behaves according to the value of the syntax element and to the values of previously decoded syntax elements. When a value of a syntax element is used in the syntax tables or the text, it appears in regular (i.e., not bold) type.

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	ITU-T Rec. H.265 (12/2016) High efficiency video coding at p. 18. **7 Syntax and semantics** **7.1 Method of specifying syntax in tabular form** The syntax tables specify a superset of the syntax of all allowed bitstreams **. . .** **. . .** **. . .** When **syntax_element** appears, it specifies that a syntax element is parsed from the bitstream **. . .** **. . .** **7.2 Specification of syntax functions and descriptors** **. . .** The following descriptors specify the parsing process of each syntax element: – ae(v): context-adaptive arithmetic entropy-coded syntax element. The parsing process for this descriptor is specified in clause 9.3. **. . .** ITU-T Rec. H.265 (12/2016) High efficiency video coding at pp. 30-31. **7.3.8.6 Prediction unit syntax**

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	(see below)

prediction_unit(x0, y0, nPbW, nPbH) {	**Descriptor**
if(cu_skip_flag[x0][y0]) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	ae(v)
} else { /* MODE_INTER */	
merge_flag[x0][y0]	ae(v)
if(merge_flag[x0][y0]) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	ae(v)
} else {	
if(slice_type = = B)	
inter_pred_idc[x0][y0]	ae(v)
if(inter_pred_idc[x0][y0] != PRED_L1) {	
if(num_ref_idx_l0_active_minus1 > 0)	
ref_idx_l0[x0][y0]	ae(v)
mvd_coding(x0, y0, 0)	
mvp_l0_flag[x0][y0]	ae(v)
}	
if(inter_pred_idc[x0][y0] != PRED_L0) {	
if(num_ref_idx_l1_active_minus1 > 0)	
ref_idx_l1[x0][y0]	ae(v)
if(mvd_l1_zero_flag && inter_pred_idc[x0][y0] = = PRED_BI) {	
MvdL1[x0][y0][0] = 0	
MvdL1[x0][y0][1] = 0	
} else	
mvd_coding(x0, y0, 1)	
mvp_l1_flag[x0][y0]	ae(v)
}	
}	
}	
}	

ITU-T Rec. H.265 (12/2016) High efficiency video coding at p. 55.

7.4.9.6 Prediction unit semantics

. . .

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS

	merge_idx[x0][y0] specifies the merging candidate index of the merging candidate list where x0, y0 specify the location (x0, y0) of the top-left luma sample of the considered prediction block relative to the top-left luma sample of the picture.
	
	When merge_idx[x0][y0] is not present, it is inferred to be equal to 0.
	
	ITU-T Rec. H.265 (12/2016) High efficiency video coding at p. 107.
	
	8.5.3.2.2 Derivation process for luma motion vectors for merge mode
	
	. . .
	
	Outputs of this process are as follows, with X being 0 or 1:
	
	. . .
	
	- the reference indices refIdxLXA$_0$, refIdxLXA$_1$, refIdxLXB$_0$, refIdxLXB$_1$ and refIdxLXB$_2$ of the neighbouring prediction units,
	
	. . .
	
	- the motion vectors mvLXA$_0$, mvLXA$_1$, mvLXB$_0$, mvLXB$_1$ and mvLXB$_2$ of the neighbouring prediction units.
	
	. . .
	
	5. The merging candidate list, mergeCandList, is constructed as follows:
	
	i = 0
	if(availableFlagA$_1$)
	mergeCandList[i++] = A$_1$
	if(availableFlagB$_1$)
	mergeCandList[i++] = B$_1$
	if(availableFlagB$_0$)
	mergeCandList[i++] = B$_0$

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS

	if(availableFlagA_0)
	mergeCandList[i++] = A_0
	if(availableFlagB_2)
	mergeCandList[i++] = B_2
	if(availableFlagCol)
	mergeCandList[i++] = Col
	
	. . .
	9. The following assignments are made with N being the candidate at position merge_idx[xOrigP][yOrigP] in the merging candidate list mergeCandList (N = mergeCandList[merge_idx[xOrigP][yOrigP]]) and X being replaced by 0 or 1:
	refIdxLX = refIdxLXN (8-120)
	predFlagLX = predFlagLXN (8-121)
	1. When use_integer_mv_flag is equal to 0 and the reference picture is not the current picture, the following applies:
	mvLX[0] = mvLXN[0] (8-122)
	mvLX[1] = mvLXN[1] (8-123)
	. . .
	
	ITU-T Rec. H.265 (12/2016) High efficiency video coding at pp. 147-149.
	
	9 Parsing process
	9.1 General
	Inputs to this process are bits …
	Outputs of this process are syntax element values.

EXHIBIT 9
UNITED STATES PATENT NO. 10,536,714
CLAIM CHART FOR INFRINGEMENT OF CLAIM 9 BY HISENSE ACCUSED PRODUCTS

U.S. PATENT NO. 10,536,714	HISENSE ACCUSED PRODUCTS
	This process is invoked when the descriptor of a syntax element in the syntax tables in clause 7.3 is equal to ue(v), se(v) (see clause 9.2), or ae(v) (see clause 9.3). ITU-T Rec. H.265 (12/2016) High efficiency video coding at p. 198. **9.3 CABAC parsing process for slice segment data** **9.3.1 General** This process is invoked when parsing syntax elements with descriptor ae(v) in clauses 7.3.8.1 through 7.3.8.11. Inputs to this process are a request for a value of a syntax element and values of prior parsed syntax elements. Output of this process is the value of the syntax element. ITU-T Rec. H.265 (12/2016) High efficiency video coding at p. 201.